Formulating LUTI Calibration as an Optimisation Problem: Example of Tranus Shadow Price Estimation

Thomas Capelle, Peter Sturm, Arthur Vidard

Available online at www.sciencedirect.com
ScienceDirect
Procedia Engineering 00 (2015) 1–8
Procedia Engineering
www.elsevier.com/locate/procedia

Symposium "Towards integrated modelling of urban systems"

Formulating LUTI Calibration as an Optimisation Problem: Example of Tranus Shadow Price Estimation

Thomas Capelle[a,b,c,*], Peter Sturm[a,b,c], Arthur Vidard[a,b,c]

[a] Inria
[b] Univ. Grenoble Alpes, LJK, F-38000 Grenoble, France
[c] CNRS, LJK, F-38000 Grenoble, France

Abstract

Much of the planet's energy consumption, pollutions, waste generation etc. happens in cities., which makes it important to consider urban areas in efforts aiming at sustainable development. Transportation and land use planning has become essential as a decision aid tool. Development of LUTI models (Land Use and Transportation Integrated models) has increased during the last 20 years. Calibration of large-scale LUTI models is a challenging task. It is usually partitioned into a set of smaller, partial parameter estimation problems of individual components of a model, and an integrated calibration of the composite model, taking into account the mutual interactions between these components, is most often lacking. This work presents a reformulation of the calibration of the Tranus model as an optimisation problem. This methodology is applied to the estimation of the endogenous variable called "shadow prices", this variable acts as an error term of the localisation utility function. We also present a test methodology for validating the calibration against synthetic data that perfectly fit observations. We present this methodology on a small example that permits us to get a visual assessment of the solution.

Keywords: Land Use, Optimisation, Transport, LUTI

1. Introduction

Integrated land use and transport modelling has attracted the attention of researchers since 1960 [1]. Over the years, a large number of models have come into existence. It is well known that integration of land use and transport models creates a complex nonlinear system, which evolves in different scales. Analysing these complex systems is typically a hard problem, especially in the presence of uncertainty, whose effects may be difficult to assess. The interaction between all the components of a model makes that small changes

*Corresponding author
Email addresses: thomas.capelle@inria.fr (Thomas Capelle), peter.sturm@inria.fr (Peter Sturm), arthur.vidard@inria.fr (Arthur Vidard)

in one parameter can cause large changes in the model output. In such cases, calibration plays a central role, as it helps us determine optimal parameters. Calibrating this type of models is a process that requires many steps. Also, the data needed are not always readily available, but even if so, finding the set of parameters that best replicate the data is not simple.

This work is an effort toward an automatic calibration of Tranus, a widely used open-source LUTI model [2]. Formulating the calibration of a LUTi model as an optimisation problem has been done before, for example for the MEPLAN model [3], but the complex nature of each model may force the use of a tailored solution. We propose an optimisation framework for the calibration of Tranus, particularly for obtaining a good estimation of the shadow-price variables.

2. Methodology

2.1. Description of Tranus

Tranus [2] provides a generic framework to model land use and transportation in an integrated manner, both on urban and regional levels. The region of interest is divided into *economic sectors* and *spatial zones*, generalising the classical input-output model proposed by Leontief [4]. Then Tranus combines two modules: the *land use and activity* module which simulates a spatial economic system by assessing the activity locations and economic sector interactions; and a *transportation* module, which estimates the use of the transport network and the associated disutility.

The two modules in the system use discrete choice logit models [5], linked together in a consistent way. This includes activity-location, land-choice, and multi-modal path choice and assignment. The modules are then run iteratively, such that production and consumption demands for each zone are met and equilibrium is achieved. A detailed description of the equations of Tranus land use and transportation modules can be found in [2].

2.2. The activity and land use module

The land use module's objective is to find an equilibrium between the production and demand of all economic sectors and spatial zones of the modelled region.

The equilibrium between these depends further on various economic parameters that aim at representing the behaviour of people and businesses, such as demand elasticities and variables representing the general attractiveness of zones (beyond land rent). **Productions** X_i^n express how many "items" of each economic sector n are present in each zone i. **Demands** D_i^{mn} express how many items of a sector n are demanded by the part of sector m that is located in zone i. Finally, p_i^n defines the **price** of (one item of) sector n located (or produced) in zone i. Here, "price" is dictated by land or floorspace prices, which are true prices, whereas the "price" of a household (roughly speaking, its demand for salary) is derived from the floorspace occupied by the household (see for instance [6]).

All these variables are computed from one another by a system of about a dozen equations, see [2] for details. Since they depend on one another (for instance demand generates production and vice-versa), we are in the presence of a dynamic system. A sketch of the central parts of this system is shown in Figure 1, where we omit many details in order to make this paper as self-contained as possible. It shows the sequence of computations done in Tranus' land use module. At each iteration of the process, current prices are fed into the computation of demands (via intermediate variables not detailed here) which in turn are fed into the computation of productions. Given the new distribution of productions across sectors and zones, production and consumption costs are computed (marked as c in the figure), based on the current prices and transportation costs. These are then used in the next iteration to determine new prices, and the above computations are repeated. The entire process starts from floorspace/land prices, which are given by collected data, as well as productions destined for exportation outside the area of study, which are also given. It is repeated until convergence; concretely, until convergence of productions X and prices p (this implies convergence of all other variables).

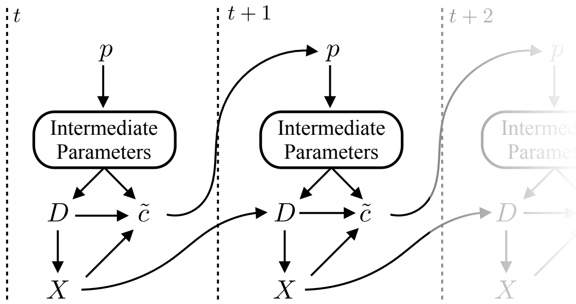

Fig. 1. Sketch of computations in the land use and activity module.

The subset of model equations relevant to this paper, is as follows. Demand is computed for all combinations of zone i, demanding (consuming) sector m and demanded sector n:

$$D_i^{mn} = (X_i^{*m} + X_i^m)\, a_i^{mn} S_i^{mn} \tag{2.1}$$

$$D_i^n = D_i^{*n} + \sum_m D_i^{mn} \tag{2.2}$$

where X_i^{*m} is the given exogenous production (for exports), X_i^m the induced endogenous production obtained in the previous iteration (or initial values), and D_i^{*n} exogenous demand. D_i^n in (2.2) then gives the total demand for sector n in zone i. a_i^{mn} is a technical demand coefficient and S_i^{mn} is the substitution proportion of sector n when consumed by sector m on zone i.

In parallel to demand, one computes the **utility** of all pairs of production and consumption zones, j and i:

$$U_{ij}^n = p_j^n + h_j^n + t_{ij}^n \ . \tag{2.3}$$

Here, t_{ij}^n represents transport disutility. Since utilities and disutilities are difficult to model mathematically (they include subjective factors such as the value of time spent in transportation), Tranus incorporates adjustment parameters h_j^n, so-called shadow prices, amongst the model parameters to be estimated.

From utilities, we compute the probability that the production of sector n demanded in zone i, is located in zone j. Every combination of n, i and j is computed:

$$Pr_{ij}^n = \frac{A_j^n exp\left(-\beta^n U_{ij}^n\right)}{\sum_h A_h^n exp\left(-\beta^n U_{ih}^n\right)} \ . \tag{2.4}$$

Here, h ranges over all zones, A_j^n represents attractiveness of zone j for sector n and β^n is the dispersion parameter for the multinomial logit model expressed by the above equation.

From these probabilities, new productions are then computed for every combination of sector n, production zone j and consumption zone i:

$$X_{ij}^n = D_i^n Pr_{ij}^n \ . \tag{2.5}$$

Total production of sector n in zone j, is then:

$$X_j^n = \sum_i X_{ij}^n \ . \tag{2.6}$$

Given the computed demand and production, consumption costs are computed as

$$\tilde{c}_i^n = \frac{\sum_j X_{ij}^n \left(p_j^n + tm_{ij}^n\right)}{D_i^n} \tag{2.7}$$

where tm^n_{ij} is the monetary cost of transporting one item of sector n from zone j to zone i.

These finally determine the new prices:

$$p^m_i = VA^m_i + \sum_n a^{mn}_i S^{mn}_i \tilde{c}^n_i \tag{2.8}$$

where VA^m_i is value added by the production of an item of sector m in zone i, to the sum of values of the input items.

2.3. Land Use Module Calibration

The calibration process consists in adjusting the model such as to reproduce the actual observed behaviour of a study area in a given timeframe or base year. It is usually performed by experts and is based on a mix of numerical parameter estimation procedures, interactive trial-and-error, and assessment of the model calibration against observed data and expert opinion. Once a LUTI model's parameters are calibrated using data from a base year (or possibly multiple base years), the usual application of a model is to use it to predict the evolution of land use and transportation usage, for different alternative scenarios of future changes in development strategies, transportation infrastructure, fiscal policies, etc.

The calibration of the land use module is usually done by a hierarchical process of the following type. The model parameters are split in three sets: *(i)* parameters that are computed independently from all others using appropriate data, *(ii)* the shadow prices h^n_j (adjustment parameters) of equation (2.3), and *(iii)* all remaining parameters. The latter two sets of parameters are estimated in an iterative process: given initial values of the third set of parameters, one estimates shadow prices for which the model, after convergence, reproduces the productions observed in the study area, denoted by $X^n_{0,i}$. Then, the remaining parameters are updated using additional observations and constraints on the shadow prices: one wants to make these as small as possible. This process is repeated interactively by the expert modeller until a compromise deemed satisfactory, is achieved between model fit, constraints, and plausibility of the estimated economic parameters.

As for the estimation of the shadow prices, a simple method is used: at the end of each iteration (cf. Figure 1 and the above equations), shadow prices are updated as follows:

$$h^{n,t+1}_i = (h^{n,t}_i + p^{n,t}_i)\frac{X^{n,t}_i}{X^n_{0,i}} - p^{n,t+1}_i \tag{2.9}$$

The rationale is to increase shadow prices if the production computed by the model exceeds observed production and vice-versa, so that in the next iteration, computed productions hopefully come closer to observed ones. So computation of shadow prices in iteration $t+1$, depends on the values of the productions and prices at iteration t, as shown in equation (2.9).

2.4. Optimisation-Based Calibration Approach

We propose a calibration approach that replaces the iterative scheme depicted in Figure 1 by an optimisation framework. To be able to do this, we have to rewrite the computation of the different equations exposed above in a compact formulation. Our objective is to present the problem via a cost function to be minimised. Doing so, enables the use of the many available numerical optimisation tools existing in the literature and in software libraries. The general problem formulation we adopt, is as follows:

- If we call Y the output of the model, and Σ the set of parameters, we can look at the model as a function of its parameters $Y(\sigma)$ with $\sigma \in \Sigma$.
- The data for a given period i is Y_i, having $i = 0$ would be the base year.
- We can define a cost function $f : \Sigma \to \mathbb{R}^+$, this function would give a real measure of how *good* is the output of the model, compared to the data for a given parameter set. For example, $f(\sigma) = d(Y(\sigma), Y_0)$ computes the distance d from the output of the model $Y(\sigma)$ against the base's year data for a given parameter σ.

With this in mind, we can write down the calibration as an optimisation problem, where the optimal solution would be the parameter $\sigma^* \in \Sigma$ that satisfies:

$$\sigma^* = \arg\min_{\sigma \in \Sigma} f(\sigma)$$

We will apply this strategy to the calibration of the land use and activity module in Tranus.

LCAL. In Tranus, the calibration of the land use and activity module is done in a program called LCAL (Land use CALibration). The objective of the calibration, is to fit the model to base year's data. For achieving this, the model has a set of parameters that can be adjusted (elasticities, attractors, discrete choice model constants, etc.). The job of the modeler, is to adjust the input parameters, to make the model fit the base year production data as closely as possible. To be sure that the model can replicate the base year's production, a correction term is added inside LCAL. This variable is added to the utility function (2.3) and acts as a correcting price, called the shadow price h. One shadow price is added for each production term. What LCAL does in the actual implementation of Tranus, is for any given set of parameters, it will iterate trough the scheme presented in Figure 1 and adjust the shadow-prices to replicate the base year X_0 for a fixed set of input parameters. This iterative algorithm tries to find a local optimum just exploiting the economic notion of the price and shadow price variable as shown in equation (2.9). This can lead to local optima, or not even find a solution.
Instead, we propose a framework, with an actual cost function to optimise and a clear and explicit description of the problem. The problem to solve in LCAL, is to adjust the shadow prices h to replicate the base year production. Written as an optimisation problem:

$$\min_h \quad f(h) = \|X(h) - X_0\|^2 \ . \tag{2.10}$$

Here, h is a vector containing all shadow prices, X_0 the vector of observed productions, and $X(h)$ the vector of productions computed by the model, after convergence of the iterative process shown in Figure 1. The dependency of these on the shadow prices is visible from equations (2.3) to (2.6).

We only have access to the productions $X(h)$ after convergence of our dynamic system of equations. Consequently, the computation of the gradient of the cost function (be it analytical or by numerical approximation) or any other variables needed by a chosen optimisation method, may be complicated or requiring waiting for convergence too. In order to solve this problem, we observe that one may cut through a loop in our dynamic system and directly compute demands and productions that are in equilibrium: in the iterative scheme shown in Figure 1, the computation of demand and production only involves equations that are linear in these parameters, cf. (2.1), (2.2), (2.5), and (2.6). These equations may be re-organised in order to form a single linear equation system in the productions and demands.

In addition, since only productions are needed in the cost function (2.10), one may reduce the problem to only estimating these (demands may be computed from estimated productions by substitution if required). To do so, we substitute equations (2.1), (2.2) and (2.5) in equation (2.6), obtaining the following linear system for the production variables X_j^n:

$$\begin{aligned}
X_j^n &= \sum_i X_{ij}^n \\
&= \sum_i D_i^n Pr_{ij}^n \\
&= \sum_i (D_i^{*n} + \sum_m a_i^{mn} S_i^{mn} (X_i^m + X_i^{*m})) Pr_{ij}^n \\
&= \underbrace{\sum_i D_i^{*n} Pr_{ij}^n + \sum_i \sum_m a_i^{mn} S_i^{mn} X_i^{*m} Pr_{ij}^n}_{\Lambda} + \underbrace{\sum_i \sum_m a_i^{mn} S_i^{mn} Pr_{ij}^n}_{\Delta} X_i^m
\end{aligned}$$

In short form:

$$X(h) = \Lambda(h, p) + \Delta(h, p)\ X(h) \tag{2.11}$$

where $\Lambda(h, p)$ and $\Delta(h, p)$ are matrices that only depend on prices and shadow prices (the other variables in the definition of these matrices, are fixed).

There remain two problems. First, in practice, the size of this linear system is too large to get an analytical solution for the productions $X(h)$. Hence, computing a gradient of the cost function, requires a finite difference approach. Still, productions computed as above are by construction in equilibrium. Second, the same is not true for the prices p: referring to Figure 1, we observe that after productions and demands are computed, prices are updated (via the computation of consumption costs $\tilde{c}$, cf. equations (2.7) and (2.8)). There is no guarantee that the recomputed prices are equal to the prices used to compute productions and demands – a constraint that is necessary for an acceptable equilibrium solution.

The first problem can be addressed easily: since we want to replicate the base's year production X_0, we can replace the production in the right side of equation (2.11) by the base's year production X_0. Hence, productions predicted by the model, can be directly computed instead of being obtained by numerically solving a linear equation system. Further, this enables the analytical computation of derivatives of the cost function, which makes its optimisation more efficient.

As for the second problem, we address it by adding the prices explicitly to the variables of the optimisation problem described in equation (2.10). Also, a term is added to the cost function, that expresses the difference between the prices computed by the model though equations (2.7) and (2.8), and the values of the price variables given as input. At convergence, this difference should be zero, signifying an equilibrium between prices and costs. The final proposed optimisation scheme is thus as follows:

$$\min_{h,p} \quad \|X(h, p, X_0) - X_0\|^2 + \|\hat{p}(h, p, X_0) - p\|^2 \ . \tag{2.12}$$

Here, $X(h, p, X_0)$ refers to the productions computed by equation (2.11): as explained, observed productions X_0 are substituted in the right hand side of that equation. Likewise, $\hat{p}(h, p, X_0)$ refers to prices computed using equations (2.7) and (2.8), again based on observed productions.

Note that all terms in the above cost function, as well as their partial derivatives, can be computed in closed-form as functions of the unknowns h and p from equations (2.4)-(2.8). This enables efficient computations of the ingredients of any least squares or other optimisation method; in our implementation, we use the Levenberg-Marquardt method [7] to solve problem (2.12).

2.5. Overview of the Complete Algorithm

The general idea for computing the shadow prices is solving the dual objective system proposed in (2.12). We separate the optimisation in two stages, between non-transportable and transportable sectors. Non-transportable sectors have to be consumed where they are produced, so there is no transportation cost involved. The later simplifies greatly the computation of the production (for instance, the location probability vanishes). Land is a non-transportable sector and must be consumed in place. For the later, the prices are know, so the optimisation problem from equation (2.12) is reduced to:

$$\min_{h} \quad \|X(h, X_0) - X_0\|^2 \ . \tag{2.13}$$

This optimisation problem is easier to solve, and it can be separated in one optimisation problem per geographical zone. This is the first step in our optimisation framework. Secondly, we compute the shadow prices and prices for the transportable sectors solving (2.12). A pseudo-code of our algorithms follows:

Algorithm 1 Shadow prices computation algorithm for non-transportable sectors

```
procedure NON-TRANSPORTABLE_OPTIMISATION(X_0, h_0)
    for each zone do
        Estimate h for all non-transportable sectors
        // price p is given
        // Minimize (2.13) starting from h_0 using least square method [7], residual and derivatives computed analytically.
    return optimal shadow price h*
```

Algorithm 2 Shadow prices computation algorithm for transportable sectors

```
procedure TRANSPORTABLE_OPTIMISATION(p_0, X_0, h_0)
    for all sectors and zone do
        Estimate h and p for all transportable sectors
        // Minimize (2.12) starting from (h_0, p_0) using least square method [7], residual and derivatives computed analytically.
    return optimal shadow price h* and corresponding equilibrium price p*
```

Algorithm 1 is executed once per geographical zone. Each of this problems is of relatively small size (number of land use sectors) and easy to solve. After the computation of non-transportable sectors, Algorithm 2 is executed. For the later, the optimisation is done for all the geographical zones and transportable sectors at the same time.

Separating the problem in two stages, simplifies the computations and enables the calibration to be carried out for the non-transportable sectors before moving to the transportable sectors. This is only possible due to the nature of transportable and non-transportable sectors and the way how they interact, more details in [2].

2.6. Test data

For LUTI models it is notoriously difficult to evaluate a calibration, due to the difficulty of obtaining ground truth information on estimated parameters. To be able to test our optimisation scheme, we constructed a set of input data (X_0, $parameters$) that has a perfect calibration (up to round-off error). Concretely, we define shadow price values that we want to be our ground truth (usually, we define them as equal to zero). We then solve a sub-problem of the calibration problem (2.12), where observed productions are no longer considered. The computation of prices in equilibrium is the only thing that we need. As we don't want to reproduce the base's year productions, we iterate (2.7) and (2.8) until a price is found. Upon convergence of this simpler problem, we use the productions computed using $h = 0$ and the computed p, as ground truth productions X_0 of a synthetic test data set. We thus obtain test data that are close to real data (we start by initialising using observed productions) and for which we know a globally optimal equilibrium solution. This is a practical for two reasons. First, we are sure to know the optimum set of shadow prices. Secondly, it permits assessing the robustness of the optimisation algorithm, testing different starting points of the algorithm and checking if the convergence to the optimal value is attained.

3. Results

In terms of optimisation, the functions Δ and Λ are differentiable, even class C^∞, so we develop an optimisation framework using the optimisation algorithm presented in [7]. The Jacobian of the function is computed analytically.

3.1. Example:

We tried a small example, based in the *Example_C* from the Tranus website. We generated the *perfectly fitted* model, with shadow prices $h_i^n = 0$ for each sector n and zone i. This example, has 3 zones and 5 sectors. We did some cuts near the optimal value, and plotted the function $f(h, p)$. In Figure 2, the graph set is generated evaluating the function f for shadow-prices h of the first sector and the three different zones. As expected, the cost function is zero at $h = 0$, and increases its value when we get far away from the optima. The function appears to be locally convex near the optimal value (at least for these 3 parameters)

What is really interesting, is to have a plot of the function f near the optimum, for a fixed sector and zone, and see how the function behaves with h and p. If we consider sector 1 and zone 1, we can plot f near the optimal value $(h_1^{*1}, p_1^{*1}) = (0, 2.676)$ as shown in the last graph of Figure 2. Here we can observe that as the shadow price gets larger the value of f increases up to a plateau state ($X_1^1(h) \to 0$ and then $f(h, p)$ becomes constant). In the case of the price p, if we move away from the optimal value $p = 2.676$, the cost increases quadratically.

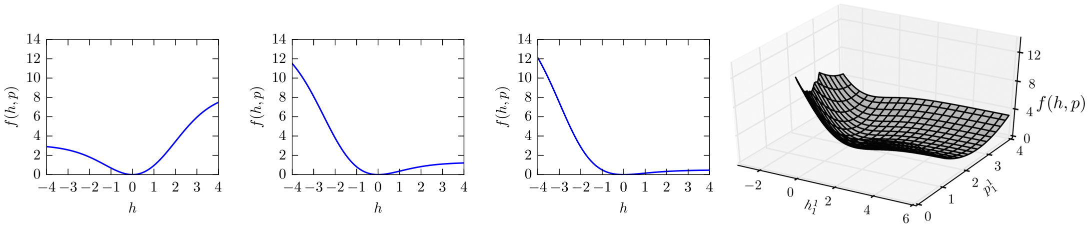

Fig. 2. First three graphs: Slices function f along h_i^1, for each zone $i \in [1, 2, 3]$. Last graph: Plot of function f for a given pair (h_1^1, p_1^1) near the optimal value ($h_1^{*1} = 0$)

We tested the robustness of the optimisation scheme with 1000 random initial values, always converging to the same optimum. The prices are in the interval $[0, 4]$, so considering the shadow prices in $[-10, 10]$ is highly representative.

3.2. North Carolina Model:

We tested our optimisation methodology in a North Carolina model with 10 economic sectors and 101 geographical zones. The algorithm proved to converge fast and to the same optimal value whatever initial solution was given. This is a scenario constructed and calibrated with the actual implementation in Tranus, so both model converge, however the Tranus implementation is more sensible to the starting point. We tested random starting points for both prices and shadow prices. From 1000 different starting points, the Tranus implementation converged in 66 cases (6.6%) and the optimisation algorithm converged in 100% of the cases. Anyhow, this is not a full representation of the whole spectrum of models, and further investigation is needed. Actually, this methodology would be very useful for a modeller calibrating a new model, whose parameters are unknown. In an early model, having convergence and actual output is often lacking. Even a joint approach could be conceived, where we begin with a global optimisation algorithm, and finish with an iterative process similar to the one implemented in Tranus.

4. Conclusions

We have a developed a optimisation methodology that give us a partial calibration of the shadow price parameters of Tranus. Secondly, we have proposed a technical contribution to the way equilibrium equations are handled in Tranus, allowing the use of sophisticated optimisation techniques. Finally, the procedure of generating synthetic data files for testing the calibration methodology is a practical way to test and check the performance of a general calibration scheme.

Acknowledgment: This work is supported by the CITiES project (ANR-12-MONU-0020).

References

[1] M. Wegener, Overview of land-use transport models, in: D. A. Hensher, K. Button (Eds.), Transport Geography and Spatial Systems, Handbook in Transport, Pergamon/Elsevier Science, Kidlington, UK, 2004, pp. 127–146.

[2] T. de la Barra, Mathematical description of TRANUS, Tech. rep., Modelistica, Caracas, Venezuela (1999). URL `http://www.tranus.com/tranus-english`

[3] J. E. Abraham, J. D. Hunt, Parameter estimation strategies for large-scale urban models, Transportation Research Record: Journal of the Transportation Research Board 1722 (1) (2000) 9–16.

[4] W. Leontief, A. Strout, Multi-Regional Input-Output Analysis, Structural Interdependence and Economic Development, London: Mcmillan, 1963.

[5] D. McFadden, K. Train, Mixed MNL models for discrete response, Journal of Applied Econometrics 15 (2000) 447–470.

[6] I. Lowry, A model of metropolis, Tech. Rep. Memorandum RM-4035-RC, The RAND Corporation, Santa Monica, California (1964).

[7] K. Levenberg, A method for the solution of certain non-linear problems in least squares, Quarterly of Applied Mathematics 2 (1944) 164–168.